

# design ideas

Edited by Bill Travis and Anne Watson Swager

## Shift register makes a fast counter

Yoram Stern, ECI Telecom, Petach-Tikva, Israel

**A**SIC DESIGNS COMMONLY require a fast, synchronous counter.

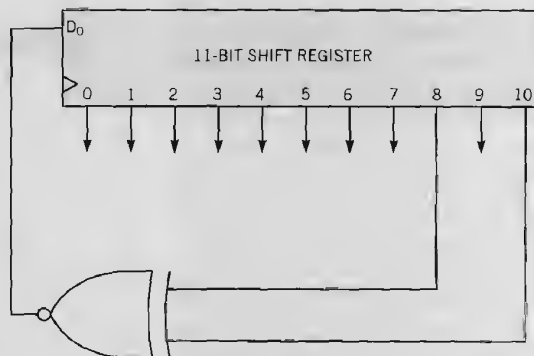
The carry propagation is the limiting factor in a fast-counter implementation. In a binary synchronous counter, a logic function of the value of all the bits in the counter determines the value of the most significant bit in the next clock edge. When the counter is long and its frequency is high, this function evaluation can take longer than one clock cycle.

One of the methods you can use to overcome this difficulty is a linear-feedback shift-register (LFSR) technique. You can build an LFSR with an N-bit shift register with XNOR or XOR feedback from the last output,  $Q_N$ , to the serial input  $D_0$  (Figure 1). An LFSR of N bits counts to  $2^N - 1$ ; a binary counter counts to  $2^N$ .

An LFSR has several advantages:

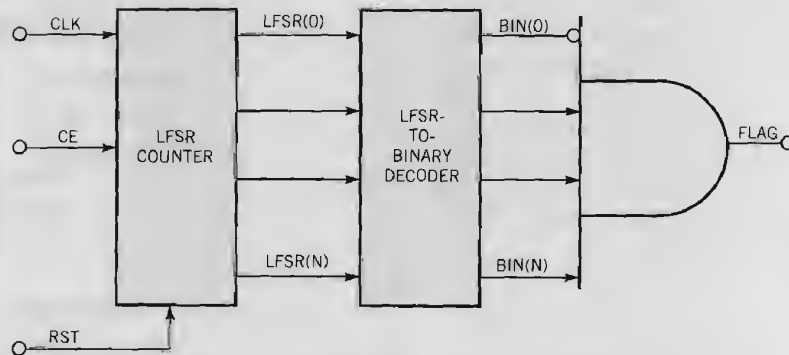
Figure 1

N	XNOR FEEDBACK FROM OUTPUTS
4	3, 2
5	4, 3
6	5, 4
7	6, 5
8	7, 5, 4, 3
9	8, 4
10	9, 6
11	10, 8
12	11, 5, 3, 0
13	12, 3, 2, 0
14	13, 4, 2, 0
15	14, 13
16	15, 14



An N-bit shift register with XNOR or XOR feedback from the last output,  $Q_N$ , to the serial input,  $D_0$ , produces an LFSR counter.

Figure 2



NOTE: WHEN  $N=4$ , THE COUNTER ASSERTS THE FLAG SIGNAL IN THE 14TH CLOCK ( $BIN="1110"$ ).

Adding a decoder after the LFSR translates the LFSR's pseudorandom count to a regular binary count.

Shift register makes a fast counter.....95

RS-232C monitor operates without a power supply.....98

Simple circuits control RC servos.....100

One amplifier makes one-shot.....102

High-power latch provides 48V, 10A.....104

Tiny IC debounces pushbutton switch.....106

Back-to-back FETs thwart reverse current.....106

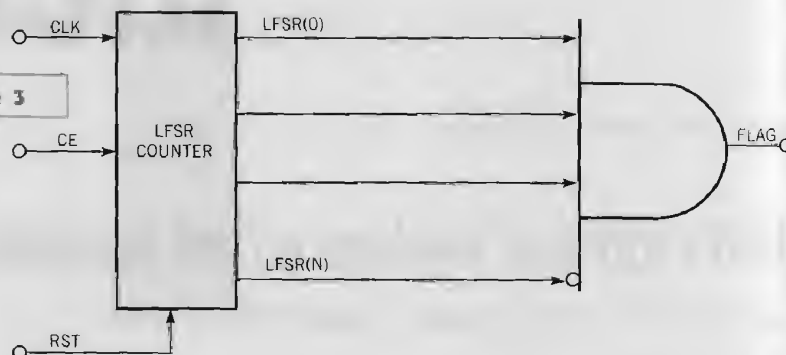
tages over a binary counter: An LFSR is much faster because the most complex logic function that needs evaluation for the next state is a four-input XOR, and LFSR logic is smaller than binary-counter logic.

An LFSR also has some disadvantages. The count of the LFSR is not intuitive. If you need a control in the 16th count, it is more intuitive to look for "10000" in a binary counter than to look for a strange "11101" in an LFSR. Furthermore, when you watch the waves on a logic simulator, you feel more relaxed when the counter is incrementing in a binary intuitive manner than a strange LFSR jump.

To overcome this disadvantage, you can use a decoder following the LFSR, which translates the LFSR pseudorandom count to a regular binary count (Figure 2). A C program consists of VHDL code that implements the LFSR-to-binary decoder (Listing 1). (You can download the program from EDN's Web site, [www.ednmag.com](http://www.ednmag.com). At the registered-user area, go into the Software Center to download the file from DI-SIG, #2345.) The additional logic for the decoder is no burden, because any logic-synthesis tool eliminates it. The silicon implementation consists of the LFSR followed by a somewhat-bizarre combination of AND gates and inverters (Figure 3).

Counters often divide fast clocks to make slower clocks or clock enables. Such clock dividers usually require a counter with a binary count because it is necessary to divide the clock by powers of 2. LFSRs are unsuitable for this application because they count by  $2^N - 1$ . To circumvent this disadvantage, shorten the count sequence to get the desired modulus by synchronously resetting the LFSR counter after decoding the appropriate count. Although this scheme sacrifices the speed of the LFSR, an LFSR is still much faster than a regular binary counter. (DI #2345)

Figure 3



NOTE: WHEN  $N=4$ , THE COUNTER ASSERTS THE FLAG SIGNAL IN THE 14TH CLOCK (BIN="1110," LFSR="1000").

After logic synthesis, the decoder that follows the LFSR consists of a rather bizarre combination of AND gates and inverters.

## LISTING 1—LFSR-TO-BINARY-DECODER C PROGRAM

```
#include <stdio.h>
#include <math.h>

#define xnor(A, B) ((A) != (B) ? (0) : (1))

// N is the number of bits in the counter
#define N 5

/* main program */

main()
{
    short int cs_lfsr[N], ns_lfsr[N];
    int cnt = 0;
    int i, max_count;

    max_count = (int) pow(2, N) - 2;

    for (i = 1; i <= N; i++)
        cs_lfsr[i] = 0;

    printf("\n case (lfsr_cnt) is");

    for (cnt = 0; cnt <= max_count; cnt++)
    {
        printf("\n when ");
        for (i = 1; i <= N; i++)
            printf("%d", cs_lfsr[i]);

        printf("\n ==> cnt <= CONV_STD_LOGIC_VECTOR(%d, %d)", cnt, N);

        // calculate ns_lfsr

        for (i = 1; i <= N; i++)
        {
            switch (i) {
                case 1 : ns_lfsr[i] = xnor(cs_lfsr[3], cs_lfsr[5]);
                        break;
                default : ns_lfsr[i] = cs_lfsr[i-1];
                        break;
            }
        }

        // move ns_lfsr to cs_lfsr

        for (i = 1; i <= N; i++)
            cs_lfsr[i] = ns_lfsr[i];

        printf("\n end case;\n");
    }
}
```

TO VOTE FOR THIS DESIGN,  
CIRCLE NO. 313

# RS-232C monitor operates without a power supply

Michele Frantisek, Brno, Czech Republic

**D**URING DEVELOPMENT WORK, you can use a simple circuit to monitor RS-232C receiving and transmitting data (Figure 1). The circuit consists of two bidirectional switches that each contain two antiparallel-connected optical couplers and two diodes that distribute the signal with the correct polarity to the optocouplers. The  $IC_1/IC_2$  switch connects to the TD line of the RS-232C link, and the  $IC_3/IC_4$  switch connects to the RD line of the link. A negative voltage or low logic level at Pin 2 of the monitor connector activates the first switch. In this case, the circuit monitors the data at the TD line. A positive volt-

age or high logic level at Pin 2 of the monitor connector activates the other switch to monitor the RD line. The serial connection of the optocoupler LEDs in both switches reduces the current that the circuit draws from Pin 2 of the monitor connector.  $R_1$  limits this current to approximately 2 mA, so this circuit needs no power supply to monitor an RS-232C link.

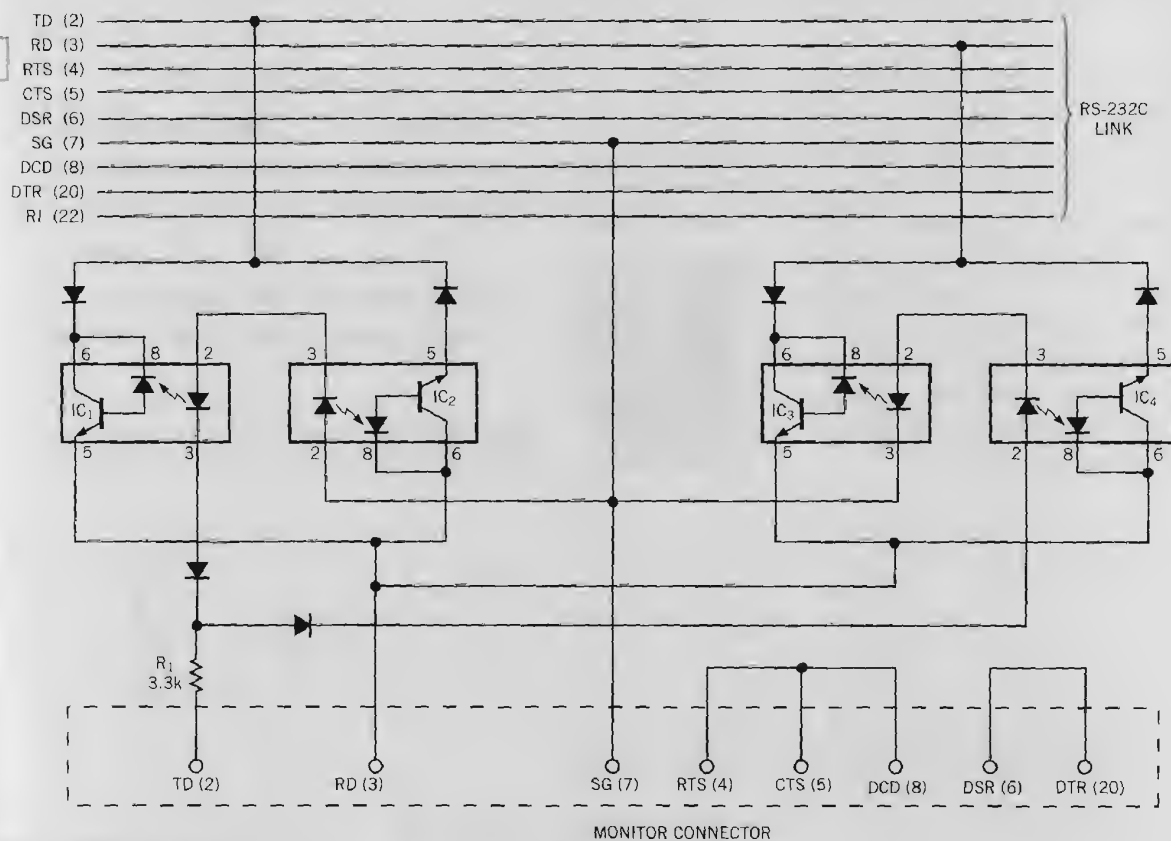
The monitor-connector configuration is equivalent to a null modem. You can attach a  $\mu P$ , for example, to monitor either the TD or the RD line of the link when the monitor device's baud rate is the same as this link. Full-duplex communication

allows the circuit to monitor both TD and RD lines when the baud rate of the monitor device is greater than or equal to double the RS-232C link's baud rate.

The 6N136 optocoupler has good parameters, especially high speed and low drive current, for this circuit. Many other types of optocouplers require a lower value for  $R_1$ , which places a greater load on the TD line of the monitor device and may cause the circuit to stop monitoring. (DI #2343)

TO VOTE FOR THIS DESIGN,  
CIRCLE NO. 314

Figure 1



NOTES:  $IC_1$  THROUGH  $IC_4=6N136$ .

$\nabla=1N4148$ .

To monitor the TD line of an RS-232C link, a low level at Pin 2 of the monitor connector activates the  $IC_1/IC_2$  switch. To monitor the RD line, a high level at Pin 2 of the monitor connector activates the  $IC_3/IC_4$  switch.

# Simple circuits control RC servos

Larry Korba, National Research Council, Ottawa, ON, Canada

**D**EVELOPERS OF ROBOTIC DEVICES sometimes need to actuate rotational elements between two positions over a range of several tens of degrees. Such motion control is particularly useful for opening doors, actuating valves, or controlling the movement of small robotic arms or legs. Two alternatives to produce this type of actuation are a solenoid and a motor attached to a gear train. Solenoids require careful mechanical design to effect the conversion from linear to rotational motion

and to achieve consistent position control between two positions. Motor-based systems require careful positioning of limit switches. An alternative to these two approaches, embodied in the circuits of **Figure 1**, involves the use of radio-control (RC) servomechanisms. Common RC servo devices are available in a variety of size, speed, and torque specifications. The range of motion of a typical RC servo is 190° (**Reference 1**). Because these servos must fit into RC cars, airplanes, and boats, they are usually small—especially considering their torque. The angular position of the servomotor shaft is related to the pulse width of an input control signal.

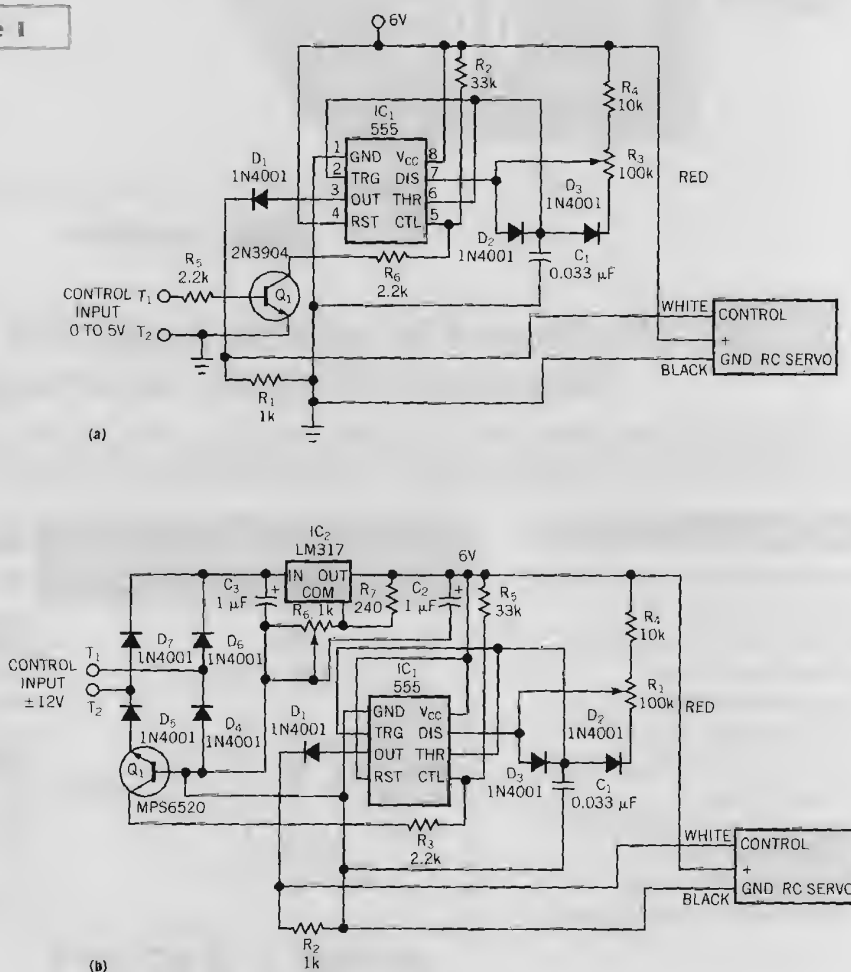
A technique for controlling the output of an RC servo between two positions uses a 0 to 5V digital signal (**Figure 1a**). The circuit requires a 5 to 6V power source.  $D_1$ ,  $D_2$ ,  $R_3$ ,  $R_4$ , and  $C_1$  configure the 555 IC to operate in astable mode at a frequency of approximately 285 kHz.  $R_3$  controls the pulse width of the output data stream over 0.3 to 2.5 msec. This setting sets the maximum counterclockwise position of the servo-

motor with the control input set at 0V. When the control input is at 5V, the output pulse train contains a 75- $\mu$ sec pulse. This pulse sets the servomotor at its most clockwise position. To calibrate the servo using the circuit in **Figure 1a**, you first set the control input at 5V. With this input, the shaft of the servo reaches its maximum clockwise position (which the ratio of  $R_2$  to  $R_2 + R_6$  controls). With the

control input set at 0V,  $R_3$  controls the rotational limit of the actuated assembly in the counterclockwise direction.

The circuit in **Figure 1b** controls the movement of an RC servo between two preset positions, using an input signal of +12 to -12V. In this case, the input control signal supplies the current that the servo and control circuit require. The bridge rectifier comprising  $D_1$  to  $D_4$  and

Figure 1



Simple circuitry uses pulse-width modulation to control an RC servomotor, using a 0 or 5V control signal (a); a self-powered version (b) uses a  $\pm 12$ V control input.

the base-emitter junction of  $Q_1$ , with the aid of some filtering by  $C_3$ , provides the input voltage for the three-terminal regulator,  $IC_2$ .  $R_6$  and  $R_7$  bias the regulator to produce a 6V output, which the servomotor requires. Transistor  $Q_1$  produces a signal of 0 or 5V, depending on the polarity of the input voltage. The circuit in **Figure 1b** supports the entry of the Carleton School Board of Ottawa (**Reference 2**) in the 1996 Canada First

Robotics Competition (**Reference 3**), in which competitors developed a remotely controlled robot that plays a specialized game of basketball. The receiver unit provides control servomotors in the robot with several bipolar outputs. (DI #2338).

#### REFERENCES

1. Jones, Joseph, and Anita Flynn, *Inspiration to Implementation*, AK Peters,

Wellesley, MA, 1993.

2. Carlton School Board Warriors Web site, [www.warriors.ottawa.com](http://www.warriors.ottawa.com).

3. Canada First Web site, [www.candafirst.org](http://www.candafirst.org).

TO VOTE FOR THIS DESIGN,  
CIRCLE NO. 315

## One amplifier makes one-shot

W Dijkstra, Waalre, The Netherlands

**A** SIMPLE CIRCUIT (**Figure 1a**) turns an amplifier into a one-shot.

At power-on, assume that the output voltage of  $IC_1$  is high. Then, the voltage across  $C_1$  increases until it is greater than the voltage at the positive input of  $IC_1$ . At this point, the output of  $IC_1$  goes low.

A short input pulse forces the output of  $IC_1$  to a high of 4.6V, and  $C_1$  begins to charge. The positive input of  $IC_1$ , near the switching low of  $IC_1$ , is equal to

$$\frac{R_1}{R_1 + R_2} \times 4.5 = 4.1V.$$

The compliance  $dV_1$  of the voltage across  $C_1$  is  $4.1 - 0.5 = 3.6V$ . The compliance of the output voltage of  $IC_1$  ( $dV_2$ ) is  $4.6 - 0.5 = 4.1V$ .

You can transform the formula

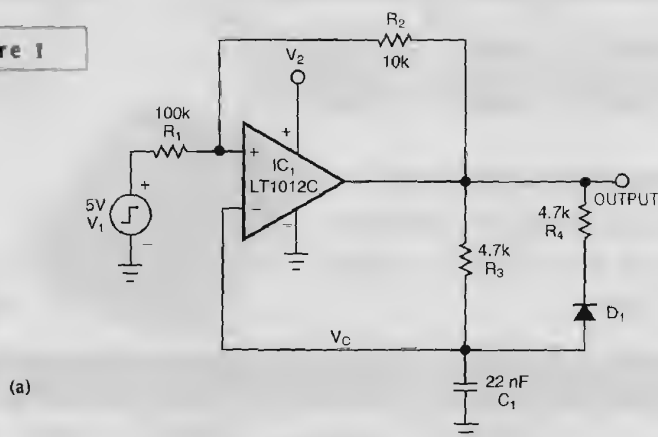
$$dV_1 = dV_2 \left( 1 - e^{-\frac{t}{R_3 \cdot C_1}} \right)$$

to the formula for the duration of the output pulse:

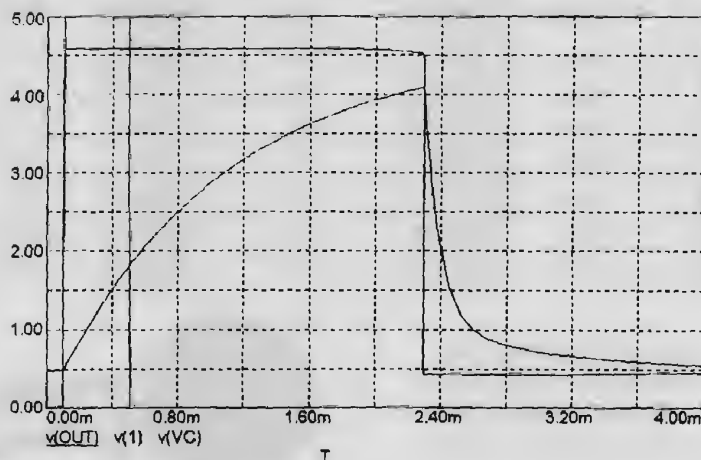
$$t = -R_3 \cdot C_1 \cdot \ln \left( 1 - \frac{dV_1}{dV_2} \right) = 2.17 \text{ mSEC.}$$

Spice analysis of the circuit (**Figure 1b**) shows accurate results.  $R_1$  and  $D_1$  ensure rapid discharging of  $C_1$ . (DI #2333)

**Figure 1**



(a)



(b)

A short input pulse forces the output of  $IC_1$  high and begins charging  $C_1$  (a). The output goes low when  $V_c$  exceeds the voltage at the amplifier's positive input (b).

TO VOTE FOR THIS DESIGN,  
CIRCLE NO. 316

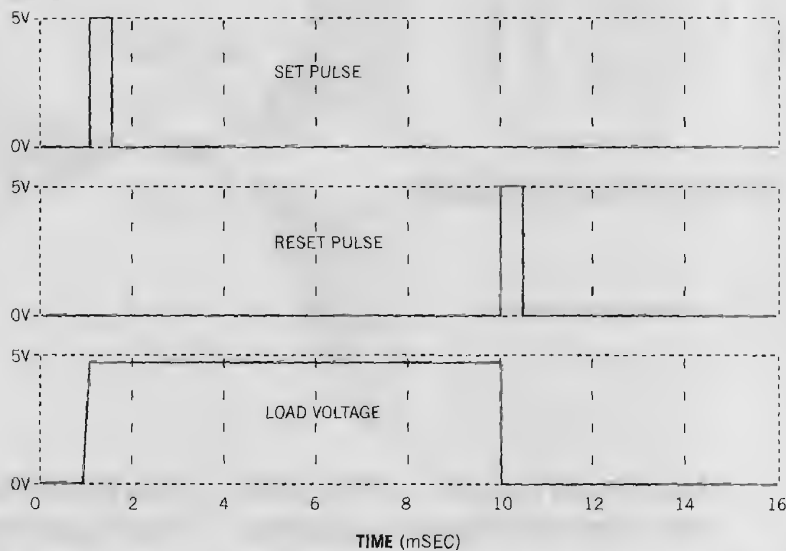
# High-power latch provides 48V, 10A

Ahmed Aboyoussef, Lucent Technologies, Holmdel, NJ

**M**ODERN CENTRAL-OFFICE telecommunications equipment has stringent reliability requirements: a bit-error rate of  $10^{-12}$  or better. The performance of the power supply is a significant factor in the quality of transmission. The supply voltage, which is nominally 48V but can reach 60V, must be within a designated range. It is standard practice to monitor the condition of the line voltage before applying power to the load. If the condition is satisfactory, you may apply power to the load via a switch. The high-power latching circuit in **Figure 1** is suitable for such applications.

The circuit sets and resets the latch with 5V pulses. International Rectifier's (www.irf.com) IRF5210 series-pass, p-channel MOSFET ( $Q_3$ ) can deliver 40A and specifies a 100V drain-source breakdown voltage. The circuit uses feedback to keep  $Q_3$  on after setting the latch. International Rectifier bases the design on MicroSim's (www.orcad.com) PSpice; **Figure 2** shows the simulation results. The circuit applies a set pulse to the gate of an IRFL4310 MOSFET ( $Q_1$ ) via the  $R_1C_1$  lowpass filter. The  $R_1C_1$  and  $R_8C_2$  lowpass filters provide immunity to EMI noise, which without filtering could set or reset the latch. The IRFL4310 is an n-channel, surface-mount SOT-223 MOSFET with 100V drain-source breakdown

**Figure 2**



This PSpice simulation shows that narrow set and reset pulses control **Figure 1**'s high-power latch.

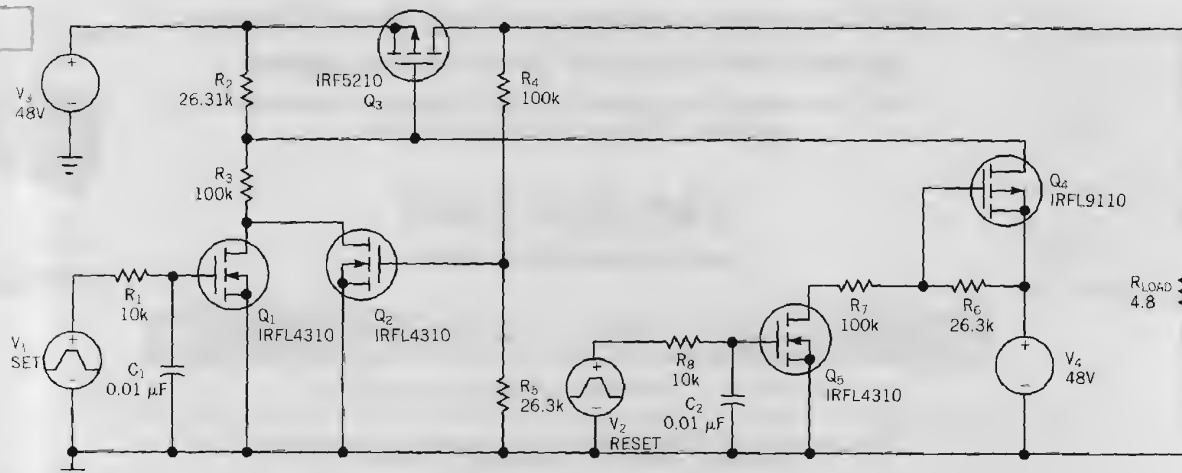
voltage. With  $Q_1$  turned on,  $Q_3$  switches on, applying the 48V line voltage to the load.

$R_4$  and  $R_5$  provide the gate-source bias voltage to turn  $Q_2$  on, thereby keeping  $Q_3$  on after termination of the set pulse. A 5V pulse is applied to the gate of  $Q_3$  via the  $R_8C_2$  filter; this action resets the latch. Activating  $Q_3$  enables the IRFL9110 p-channel MOSFET ( $Q_4$ ) to apply 48V to the

gate of  $Q_3$ , thereby disabling  $Q_3$ . The circuit's topology requires little pc-board real estate and takes advantage of the high drain-source breakdown voltages available in surface-mount, SOT-223 MOSFETs. (DI #2337).

TO VOTE FOR THIS DESIGN,  
CIRCLE NO. 317

**Figure 1**



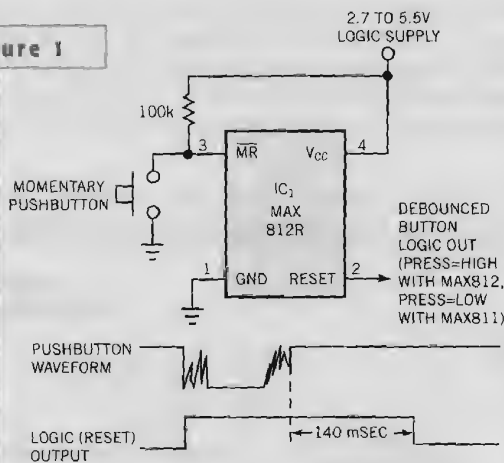
You set and reset this compact, surface-mount high-power latch circuit by applying 5V pulses to logic-level MOSFETs.

# Tiny IC debounces pushbutton switch

Len Sherman, Maxim Integrated Products, Sunnyvale, CA

**T**HOUGH THEY'RE NOT complicated, schemes for debouncing a pushbutton switch usually entail using several logic gates. It's easy to include such circuits in an ASIC. Adding a debouncer as a last-minute design change, however, can be inconvenient. In such cases, the circuit in **Figure 1** can come in handy. The circuit uses only a four-pin SO-package IC, squares up and debounces a pushbutton signal. IC<sub>1</sub> is a reset chip with a Reset output that goes high when its supply voltage drops below 2.65V or when its manual-reset input ( $\overline{MR}$ ) goes low.  $\overline{MR}$  usually connects to a system-reset input, and Reset connects to a  $\mu C$ , but the connections shown enable the de-

**Figure 1**



Using almost no real estate, this circuit squares up and debounces a pushbutton signal and extends it to at least 140 msec.

vice to debounce any signal.

The internal one-shot provides an instant response to the first falling edge on the input and then delays any further response until 140 msec after the last rising edge. The MAX812 shown inverts the pushbutton input; a similar device (MAX811) is noninverting. The ICs also incorporate a power-on reset function that asserts the reset output when  $V_{CC}$  falls below a preset threshold and also asserts the output for 140 msec after each application of power. Selecting an R-suffix part sets this threshold at its lowest value (2.63V), preventing false outputs in the debouncer except when the supply voltage fails. (DI #2340).

TO VOTE FOR THIS DESIGN,  
CIRCLE NO. 318

# Back-to-back FETs thwart reverse current

Randy Moore, Harris Semiconductor, Costa Mesa, CA

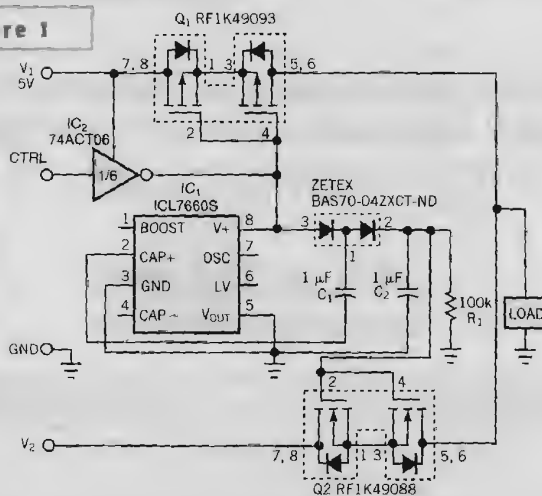
**I**T IS SOMETIMES DESIRABLE to switch a load between power sources using MOSFETs. Problems can arise when the chosen source voltage is approximately 0.6V higher than the voltage of the other source. At that point, the body diode, normally reverse-biased, becomes forward-biased and allows current flow into the source of lower voltage. The MOSFET with the forward-biased body diode can then become a cosupplier, if not the only supplier, of current to the load. You can solve the problem by adding a second MOSFET of the same channel type in the reverse direction (**Figure 1**). You connect the MOSFETs' gates in parallel. The trade-off in the scheme is that  $R_{DS(ON)}$  is twice as high as in the single-MOSFET connection.

The circuit contains two dual MOSFETs, one n-channel pair and one p-channel pair, both of

which are logic-level devices. The p-channel FETs turn on with logic 0 ( $V_{GS} \approx -5V$ ) on their gates. This action also starves the ICL7660S for power, be-

cause the IC's logic output is approximately 0V to the n-channel MOSFETs' gates. The n-channel MOSFETs, therefore, are off. Conversely, applying a logic 1 (5V) to the p-channel MOSFETs' gates, the FETs turn off, because  $V_{GS}$  is approximately 0V. The ICL7660S voltage doubler now receives power and acts as a high-side driver for the n-channel MOSFETs. The 74ACT06 is more than capable of supplying the necessary current for the ICL7660S, which consumes approximately 160  $\mu A$  while providing the gate drive for the n-channel MOSFETs. (DI #2341).

**Figure 1**



Adding two MOSFETs in a load/power switch eliminates reverse-current flow between the two power supplies.

TO VOTE FOR THIS DESIGN,  
CIRCLE NO. 319